

**AMENDMENTS TO SPECIFICATION:**

Please replace the paragraph beginning at page 3, line 19 with the following paragraph:

Therefore, the data items corresponding to the respective messages sent multiple times are queued in a queue for TASK B, while an activation request for requesting for activating TASK B is queued multiple times in an activation wait queue (Ready Queue) of the RTOS. Thereafter, when no more task (or processing, interruption) having a higher priority than TASK B exists, a series of processings for delivering messages repeats to takes place multiple times. This series of processings includes the following: (1) Activation processing for activating TASK B by RTOS; (2) Data reception processing within TASK B (Step 21); (3) Data utilization processing using the received data within TASK B (Step 22); and (4) Termination processing for terminating TASK B.

Please replace the paragraph beginning at page 6, line 27 with the following paragraph:

In the above, within the sender task (TASK A), a message is transmitted multiple times to the recipient task (TASK B) without returning the sequence to the RTOS. By contrast, within the recipient task of TASK B, a series of processings occurs every a single message. This series of processings includes the activation/termination processing (processings in the RTOS [e.g., processing for SCHEDULEs in FIG. 5]) and processing (e.g., determining processing for transferring to the processing shown in FIG. 4B). This increases a processing load in the processing unit (e.g., CPU).

Please replace the paragraph beginning at page 7, line 15 with the following paragraph:

To achieve the above object, an inter-task communications method is provided with the following. When a transmission request occurs that a data item be sent from a processing of a sender (first) task to a processing of a recipient (second) task, a data queuing is executed within the processing of the sender task. The data item is thereby stored in a queue from which the recipient task can retrieve the data item. An activation request is then outputted to an operating system for requesting for activating the recipient task. When the recipient task is activated by a processing of the operating system based on the activation request, a data ~~retrieving~~retrieval is executed within the processing of the recipient task. The data item stored in the queue is retrieved from the queue. Here, within the processing of the sender task a frequency of the activation request is provided for being less than a frequency of the data queuing. Within the data retrieving within the processing of the recipient task, more than one data item is retrieved from the queue.

Please replace the paragraph beginning at page 8, line 15 with the following paragraph:

In another aspect of the present invention, an inter-task communications method is provided with the following. When a transmission request occurs that a data item be sent from a processing of a sender (first) task to a processing of a recipient (second) task, a data queuing is executed within the processing of the sender task. The data item is thereby stored in a queue from which the recipient task can retrieve the data item. An activation request is then outputted to an operating system for requesting for activating the recipient task. When the recipient task is activated by a processing of the operating system based on the activation request, a data

retrieving is executed within the processing of the recipient task. The data item stored in the queue is retrieved from the queue. Here, when a transmission request that a given data item be sent occurs, it is determined whether the queue stores a certain data item that is being already stored before the given data item is to be stored. When the certain data item is being already stored, no activation request is then outputted. When no certain data item is being already stored, an activation request is then outputted. Within the data retrieving within the processing of the recipient task, all data items that can be retrieved from the queue ~~is~~are retrieved from the queue.